

Saveing File and Variable Attributes to Attach them to a New Variable and File

Â

[Contents](#) [Previous](#) [Next](#)

Goal:Â To save the files global attributes together with the variabla attribute to use them when writing a derived variable to a new file.

Note:Â *To run the code you need to have access to the data in the direcory '/pcmdi/IPCC'.*

Â

You can **download** and run the **python script**Â [**redecorate.py**](#) or follow the tutorial and type commands at the **cdat** or **python prompt**.

Â

Let's import the modules we will need

```
# import needed modules
import cdms, vcs, cdutil, genutil, cdtime, MA, MV
```

and open a datasetÂ we will work with

```
a=cdms.open('/pcmdi/IPCC/ipcc/data/20c3m/atm/mo/tas/gfdl_cm2_1/run1/tas_20c3m_gfdl_cm2_1_r1.xml')
```

let's import some data

```
data=a('tas')
print data.shape
```

the output is:

Â Â (1680, 90, 144)

Now lets get the files global attributes and put them into a dictionary called: file_dic

```
#
# rip out global attributes and put them in a dictionary called: file_dic
#
list_file=a.attributes.keys()
file_dic={}
for i in range(0,len(list_file)):
Â file_dic[i]=list_file[i],a.attributes[list_file[i] ]
```

let's see what's inside the list_file:

```
print list_file
```

Â Â Â Â ['comment', 'references', 'calendar', 'id', 'title', 'source', 'realization', 'cdms_filemap', 'project_id', 'cmor_version', 'institution', 'gfdl_experiment_name', 'Conventions', 'contact', 'table_id', 'experiment_id', 'directory', 'history']

```
print file_dic
```

{0: ('comment', 'GFDL experiment name = CM2.1U-D4_1860-2000-AllForc_H1. '), 1: ('references', "The GFDL Data Portal (<http://nomads.gfdl.noaa.gov/>) provides access to NOAA/GFDL's publicly available model input and output data sets. From this web site one can view and download data sets and documentation, including those related to the GFDL CM2.1 model experiments run for the IPCC's 4th Assessment Report and the US CCSP."), 2: ('calendar', 'noleap'), 3: ('id', 'none'), 4: ('title', 'GFDL CM2.1, 20C3M (run1) climate of the 20th Century experiment (20C3M) output for IPCC AR4 and US CCSP'), 5: ('source', 'GFDL_CM2.1 (2004): atmosphere: AM2.1 (am2p13fv, M45L24); ocean: OM3.1 (mom4p1p7_om3p5, tripolar360x200L50); sea ice: SIS; land: LM2; infrastructure: FMS preK release'), 6: ('realization', 1), 7: ('cdms_filemap', '[[[height,lat_bnds,lon_bnds],[[-,-,-,-,tas_A1.186101-200012.nc]] ,[[tas,time_bnds],[[0,1680,-,-,tas_A1.186101-200012.nc]]]]), 8: ('project_id', 'IPCC Fourth Assessment and US CCSP Projects'), 9: ('cmor_version', 0.9599999785420003), 10: ('institution', 'NOAA GFDL (US Dept of Commerce / NOAA / Geophysical Fluid Dynamics Laboratory, Princeton, NJ, USA)'), 11: ('gfdl_experiment_name', 'CM2.1U-D4_1860-2000-AllForc_H1'), 12: ('Conventions', 'CF-1.0'), 13: ('contact', 'GFDL.Climate.Model.Info@noaa.gov'), 14: ('table_id', 'Table A1 (20 September 2004)'), 15: ('experiment_id', 'climate of the 20th Century experiment (20C3M)'), 16: ('directory', ''), 17: ('history', 'input/atmos.186101-200012.t_ref.nc') At 12:21:55 on 01/03/2005, CMOR rewrote data to comply with CF standards and IPCC Fourth Assessment and US CCSP Projects requirements [2005-7-1 22:51:14]
 /usr/local/cdat/experimental/bin/cdscan -x tas_20c3m_gfdl_cm2_1_r1.xml tas_A1.186101-200012.nc')

and do the same with the variables attributes

```

#
# rip out data attributes and put them in a dictionary called: data_dic
#
list_data=data.attributes.keys()
data_dic={}
for i in range(0,len(list_data)):
  data_dic[i]=list_data[i],data.attributes[list_data[i] ]

```

see the list_data

```
print list_data
```

['name_in_file', 'missing_value', 'name', 'datatype', 'original_name', 'long_name', 'standard_name', 'cell_methods', 'coordinates', 'units']

and in the data_dic

```
print data_dic
```

{0: ('name_in_file', 'tas'), 1: ('missing_value', [1.00000002e+20,]), 2: ('name', 'tas'), 3: ('datatype', 'Float'), 4: ('original_name', 't_ref'), 5: ('long_name', 'Surface Air Temperature'), 6: ('standard_name', 'air_temperature'), 7: ('cell_methods', 'time: mean'), 8: ('coordinates', 'height'), 9: ('units', 'K')}

Now we can derive the annual cycle data

```

#
# =====
#
# calculate Annual Cycle
#
cdutil.setTimeBoundsMonthly(data)

```

```

#
start_time = cdtime.comptime(1979,1,1)
end_time   = cdtime.comptime(1993,12,1)
#
ac=cdutil.ANNUALCYCLE.climatology(data(time=(start_time, end_time, 'cob')))

# =====

```

And finally we can put original data attributes back on the calculated annual cycle data

```

#
# put original data attributes back on calculated annual cycle data
#
for i in range(0,len(data_dic)):
    dm=data_dic[i]
    setattr(ac,dm[0],dm[1])

```

Now we can write the data to a NetCDF file

```

#
# write out file and add global attributes to file
#
o=cdms.open('/home/hnilo/output.nc','w')
o.write(ac)
for i in range(0,len(file_dic)):
    dm=file_dic[i]
    setattr(o,dm[0],dm[1])

o.close()
a.close()

```

This tutorial was provided by [Jay Hnilo](#)

Â

[Contents](#) [Previous](#) [Next](#)